International Journal of Research in Computer Science
eISSN 2249-8265 Volume 2 Issue 4 (2012) pp. 31-38

www.ijorcs.org

A STUDY AND COMPARATIVE ANALYSIS OF CONDITIONAL RANDOM FIELDS FOR INTRUSION DETECTION

Deepa Guleria[1], M.K.Chavan[2]

[1]*PG Scholar, VPCOE Baramati*
Email: deepa.guleria@gmail.com

[2]*Asstt Professor, VPCOE Baramati*
Email:chavan_manik@yahoo.com

Abstract: *Intrusion detection systems are an important component of defensive measures protecting computer systems and networks from abuse. Intrusion detection plays one of the key roles in computer security techniques and is one of the prime areas of research. Due to complex and dynamic nature of computer networks and hacking techniques, detecting malicious activities remains a challenging task for security experts, that is, currently available defense systems suffer from low detection capability and high number of false alarms. An intrusion detection system must reliably detect malicious activities in a network and must perform efficiently to cope with the large amount of network traffic. In this paper we study the Machine Learning and data mining techniques to solve Intrusion Detection problems within computer networks and compare the various approaches with conditional random fields and address these two issues of Accuracy and Efficiency using Conditional Random Fields and Layered Approach.*

Keywords: *Intrusion Detection System, Conditional Random Fields, Network Security, Decision tree*

I. INTRODUCTION

An intrusion detection system monitors the activities of a given environment and decides whether these activities are malicious (intrusive) or legitimate (normal) based on system integrity, confidentiality and the availability of information resources. Intrusion detection as defined by the Sysadmin, Audit, Networking, and Security (SANS) institute is the act of detecting actions that attempt to compromise the confidentiality, integrity or availability of a resource [1]. Detecting intrusions in networks and applications has become one of the most critical tasks to prevent their misuse by attackers. The cost involved in protecting these valuable resources is often negligible when compared with the actual cost of a successful intrusion, which strengthens the need to develop more powerful intrusion detection systems.

There are two types of IDS depending on their mode of deployment and data used for analysis. Network Intrusion Detection Systems (NIDS) and the other is Host Intrusion Detection Systems (HIDS).NIDS monitors the packets from the network and it is an independent platform that identifies intrusion by examining the network traffic and multiple hosts. HIDS analyzes the audit data of the operation system and monitors the inbound and outbound packets from the device only. It alerts the user or administrator of suspicious activity is detected [7].Intrusion detection systems can also be classified as signature based or anomaly based depending upon the attack detection method. The signature-based systems are trained by extracting specific patterns (or signatures from previously known attacks while the anomaly-based systems learn from the normal data collected when there is no anomalous activity. The first approach is called as Misuse Detection and leads us towards Signature Based IDS while the second is called as Anomaly Detection and leads us to Behavior based IDS. The Signature based systems though have very high detection accuracy but they fail when an attack is previously unseen. On the other hand, Behavior based IDS or anomaly based may have the ability to detect new unseen attacks but have the problem of low detection accuracy [7]. Another approach for detecting intrusions is to consider both the normal and the known anomalous patterns for training a system and then performing classification on the test data. Such a system incorporates the advantages of both the signature-based and the anomaly-based systems and is known as the Hybrid System.

Hybrid systems can be very efficient, subject to the classification method used, and can also be used to label unseen or new instances as they assign one of the known classes to every test instance. This is possible because during training the system learns features from all the classes. The only concern with the hybrid method is the availability of labeled data. Further, a single system has limited attack detection coverage and it cannot detect a wide variety of attacks reliably.

We introduce hybrid intrusion detection systems based on conditional random fields which can detect a wide variety of attacks and which result in very few false alarms. To improve the efficiency of the system, we then integrate the layered framework.

II. APPROACHES TO IMPLEMENT IDS

Intrusion detection has been an active field of research for starting in 1980s after the influential paper from Anderson [7]. Several researchers have proposed various intrusion detection methods and frameworks which are available to protect a computer system or network from attacks. Various techniques such as association rules, clustering, Naïve Bayes, Support Vector Machines, Neural Networks, and others have been developed to detect intrusions. This section provides a brief literature review on these technologies and related frameworks. These methods can be broadly divided into three major categories:

A. Pattern Matching

Pattern Matching is the simple type of attack detection technique. It has the simple concept of string matching. Using pattern matching technique, IDSs generally match the text (audit records) or binary sequences against known attack signatures. A pattern matching technique basically looks for a specific attack signature which may be presented in audit record. The limitation of pattern matching approach is that it can recognize only known attacks. It requires continuous updates of attack signatures to identify new attacks. Pattern matching approach is well suited for misuse detection. Snort system is based upon pattern matching.

B. Statistical Methods

Statistical modeling is among the earliest methods used for detecting intrusions in electronic information systems. It is assumed that an intruder's behavior is noticeably different from that of a normal user, and statistical models are used to aggregate the user's behavior and distinguish an attacker from a normal user. The techniques are applicable to other subjects, such as user groups and programs. Two statistical models that have been proposed for anomaly detection: NIDES/STAT and Haystack.

C. Data Mining and Machine Learning

Data mining and machine learning methods focus on analyzing the properties of the audit patterns rather than identifying the process which generated them. These methods include approaches for mining association rules, classification and cluster analysis.

Clustering: For unsupervised intrusion detection, data clustering methods can be applied. These methods involve computing a distance between numeric features and therefore they cannot easily deal with symbolic attributes, resulting in inaccuracy. Addition, clustering methods consider the features independently and are unable to capture the relationship between different features of a single record which results in lower accuracy [9].

Data Mining: Data mining (DM), also called Knowledge-Discovery and Data Mining, is the process of automatically searching large volumes of data for patterns using association rules. Data mining approaches derive association rules and frequent episodes from available sample data, not from human experts. Using these rules, Lee et. al. developed a data mining framework for the purpose of intrusion detection[8].In particular, system usage behaviors are recorded and analyzed to generate rules which can recognize misuse attacks. The drawback of such frameworks is that they tend to produce a large number of rules and thereby, increase the complexity of the system.

Bayesian Classifiers: A Bayesian network is a model that encodes probabilistic relationships among variables of interest. This technique is generally used for intrusion detection in combination with statistical schemes, a procedure that yields several advantages, including the capability of encoding interdependencies between variables and of predicting events, as well as the ability to incorporate both prior knowledge and data. However, a serious disadvantage of using Bayesian networks is that their results are similar to those derived from threshold-based systems, while considerably higher computational effort is required.

Decision Trees: Decision trees are one of the most commonly used supervised learning algorithms in IDS due to its simplicity, high detection accuracy and fast adaptation. Decision trees used for intrusion detection select the best features for each decision node during tree construction based on some well-defined criteria [11]. One such criterion is the gain ratio which is used in C4.5.

A decision tree is composed of three basic elements:

1. A decision node specifying a test attributes.
2. An edge or a branch corresponding to the one of the possible attribute values which means one of the test attribute outcomes.
3. A leaf which is also named an answer node contains the class to which the object belongs.

Artificial Neural Networks: Neural networks are known for good performance in learning system-call sequences. Once the neural net is trained on a set of representative command sequences of a user, the net constitutes the profile of the user, and the fraction of incorrectly predicted events then measures, in some sense, the variance of the user

behavior from his profile. They can work effectively with noisy data but they require large amount of data for training and it is often hard to select the best possible architecture for the neural network [12].

Support Vector Machines: Support vector map real valued input feature vector to higher dimensional feature space through nonlinear mapping and have been used for detecting intrusions. They can provide real-time attack detection capability, deal with large dimensionality of data and perform multi class classification. Similar to the pattern matching and statistical methods, these methods assume independence among consecutive events and hence do not consider the order of occurrence of events for attack detection [17].

Markov Models: Markov chains and hidden Markov model is a set of states that are interconnected through certain transition probabilities, which determine the topology and the capabilities of the model. During a first training phase, the probabilities associated to the transitions are estimated from the normal behavior of the target system. The detection of anomalies is then carried out by comparing the anomaly score (associated probability) obtained for the observed sequences with a fixed threshold. Markov chains and hidden Markov models can be used when dealing with sequential representation of audit patterns. Hidden Markov models have been shown to be effective in modeling sequences of system calls of a privileged process, which can be used to detect anomalous traces [13]. However, modeling system calls alone may not always provide accurate classification as various connection level features are ignored. Further, hidden Markov models cannot model long range dependencies between the observations.

III. CHALLENGES AND REQUIREMENT FOR INTRUSION DETECTION SYSTEM

It is important intrusion detection must detect attacks at an early stage in order to minimize their impact. The major challenges and requirements for building intrusion detection systems are:

i. The system must be able to detect as many attacks as possible without giving false alarms i.e the system must be accurate in detecting attacks.
ii. The system must be able to handle large amount of data without affecting performance and without dropping data.
iii. A system must not only detect an attack, but also able to identify the type of attack.
iv. A system must be resistant to attacks since, a system that can be exploited during an attack may not be able to detect attacks reliably.
v. The challenge is to build a system which is scalable and can be easily customized as per the specific requirements of the environment where it is deployed.

IV. CONDITIONAL RANDOM FIELDS

A. *Conditional Probability*

Conditional probability is used to compute probability of an event Y given some other event X. Formally it is defined as:

$$P(Y \mid X) = \frac{P(X \cap Y)}{P(X)}$$

Where P(X) > 0. From this definition we can read that if the occurrence of the event X takes place in the same space as the event Y, and there are no other events that may act the occurrence of the event Y, then the conditional probability of the occurrence of the event Y given the event X is the relative proportion of outcomes that satisfy Y among those that satisfy X.

B. *Conditional Random Field Framework*

Conditional random fields [15] (CRFs) are a probabilistic framework for labeling and segmenting sequential data, based on the conditional approach described in the previous paragraph. A CRF is a form of undirected graphical model that defines a single log-linear distribution over label sequences given a particular observation sequence. The primary advantage of CRFs over hidden Markov models is their conditional nature, resulting in the relaxation of the independence assumptions required by HMMs in order to ensure tractable inference. Additionally, CRFs avoid the label bias problem [14], a weakness exhibited by maximum entropy Markov models [16] (MEMMs) and other conditional Markov models based on directed graphical models. CRFs outperform both MEMMs and HMMs on a number of real-world sequence labeling tasks.

CRF was firstly proposed by Lafferty and his colleagues in 2001, [15] whose model idea mainly came from MEMM (Maximum Entropy Markov Model).The critical difference between CRFs and MEMMs is that a MEMM uses per-state exponential models for the conditional probabilities of next states given the current state, while a CRF has a single exponential model for the joint probability of the entire sequence of labels given the observation sequence. Therefore, the weights of different features at different states can be traded off against each other. Conditional models are probabilistic systems that are used to model the conditional distribution over a set of random variables. Such models have been extensively used in the natural language processing tasks. Conditional models offer a better framework as they do not make any unwarranted assumptions on the observations and can be used to model rich overlapping features among the visible observations [6].

Lafferty, McCallum and Pereira define a CRF on observations X and random variables Y as follows:

Let X be the random variable over data sequence to be labeled and Y the corresponding label sequence.

In addition, let $G=(V,E)$ be a graph such that Y is indexed by the vertices of G. Then, (X,Y) is a CRF, when conditioned on X, the random variables Y_v obey the Markov property with respect to the graph:

$$p(Y_v \mid X, Y_w, w \neq v) = p(Y_v \mid X, Y_w, w \sim v)$$

where w ~ v means that w and v are neighbors in G, i.e., a CRF is a random field globally conditioned on X. For a simple sequence (or chain) modeling, as in our case, the joint distribution over the label sequence Y given X has the following form:

$$p_\theta(y|x) \alpha \exp\left(\sum_{e\in E,k} \lambda_k f_k(e, y|_e, x) + \sum_{v\in V,k} \mu_k g_k(v, y|_v, x)\right), \quad (1)$$

where x is the data sequence, y is a label sequence, and $Y|_S$ is the set of components of y associated with the vertices or edges in subgraph S. In addition, the features f_k and g_k are assumed to be given and fixed. Further, the parameter estimation problem is to find the parameters $\theta = (\lambda_1, \lambda_2, ...; \mu_1, \mu_2....)$ from the training data $D = (x^i, y^i)_{i=1}^N$ with the empirical distribution $p(y,x)$

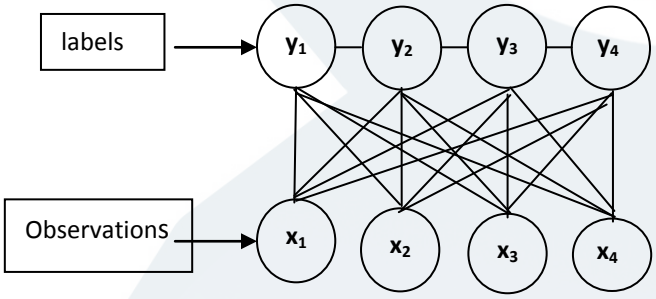
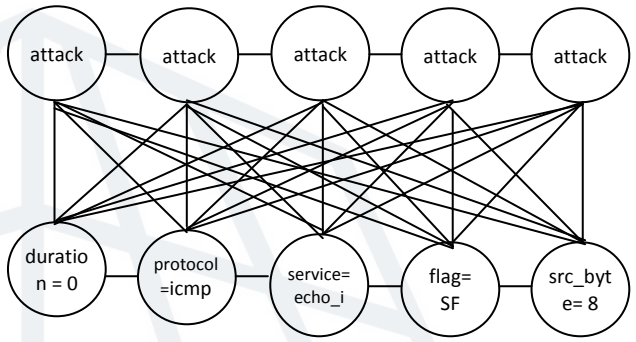
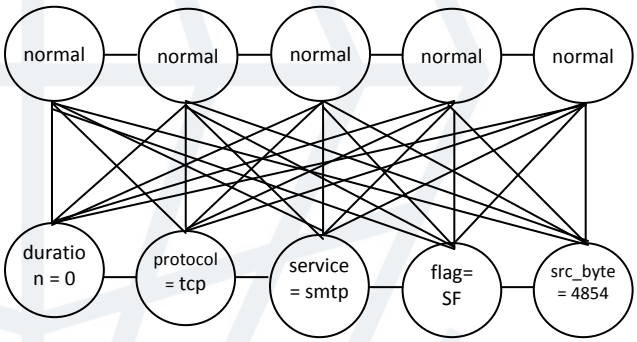


Figure 1: Graphical Representation of a CRF

The graphical structure of a conditional random field is represented in Figure1 where *x*1, *x*2, *x*3, *x*4 represents an observed sequence of length four and every event in the sequence is correspondingly labeled as *y*1, *y*2, *y*3, *y*4.The prime advantage of conditional random fields is that they are discriminative models which directly model the conditional distribution $p(y \mid x)$.Generative models such as the Markov chains, hidden Markov models and joint distribution have two disadvantages. First, the joint distribution is not required since the observations are completely visible and the interest is in finding the correct class which is the conditional distribution $p(y \mid x)$.Second, inferring conditional probability $p(y \mid x)$ from the joint distribution, using the Bayes rule, requires marginal distribution *p*(*x*) which is difficult to estimate as the amount of training data is limited and the observation *x* contains highly dependent features. As a result strong independence assumptions are made to reduce complexity. This results in reduced accuracy.



(a) Attack event



(b) Normal event

Figure 2: Conditional Random Fields for Network Intrusion Detection

In the figure 2, observation features 'duration', 'protocol', 'service', 'flag' and 'source bytes' are used to discriminate between (att) *attack* and (nor) *normal* events. The features take some possible value for every connection which are then used to determine the most likely sequence of labels < *attack*, *attack*, *attack*, *attack*, *attack* > or < *normal*, *normal*, *normal*, *normal*, *normal* >. During training, feature weights are learnt and during testing, features are evaluated for the given observation which is then labeled accordingly. It is evident from the figure that every input feature is connected to every label which indicates that all the features in an observation determine the final labeling of the entire sequence. Thus, a conditional random field can model dependencies among different features in an observation. Present intrusion detection systems do not consider such relationships.

The task of intrusion detection can be compared to many problems in machine learning, natural language processing, and bioinformatics. The CRFs have proven to be very successful in such tasks, as they do not make any unwarranted assumptions about the data. Hence, we explore the suitability of CRFs for building efficient and accurate intrusion detection.

C. Inference in CRF

For general graphs, the problem of exact inference in CRFs is intractable. However there exist special cases for which exact inference is feasible:

- If the graph is a chain or a tree, message passing algorithms yield exact solutions. The algorithms used in these cases are analogous to the forward-backward and Viterbi algorithm for the case of HMMs.
- If the CRF only contains pair-wise potentials and the energy is submodular, combinatorial min cut/max flow algorithms yield exact solutions.

D. Detecting network intrusions using layered Approach

Researchers are motivated to propose different approaches seeing the low detection rates caused by the imbalanced network intrusion dataset. Current research work proposes a staged or layered approach to detect network intrusions efficiently. The recent research work of Gupta and Nath [6], considered the attack categories as layers and different features were selected for each layer. The dataset was, therefore, divided into five attack categories for training and testing purposes of each layer. The test data passed through the cascaded layers to determine the category

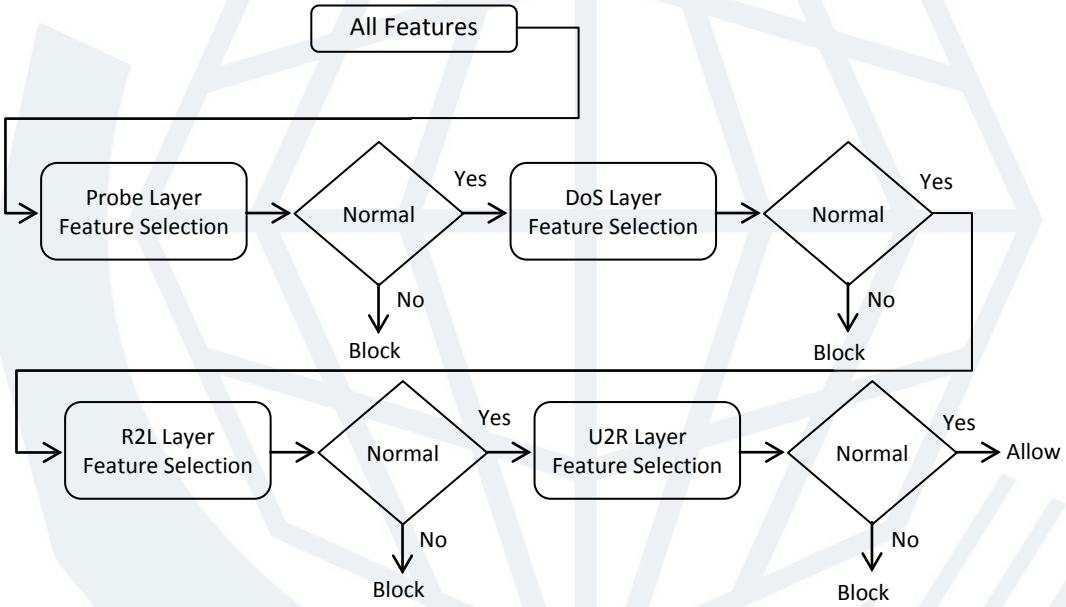


Figure 3: Integrating the Layered Framework

a record that belonged to Conditional Random Fields (CRFs) were used in the layered approach as proposed by the researcher [6]. The three layer system to ensure complete security viz. availability, confidentiality and integrity, each layer corresponding to one aspect of security. In the system, every layer is trained separately with the *normal* instances and with the *attack* instances belonging to a single attack class. Here the features involved were different in each layer. Explanation of which features should be used or not be used was given in the paper. However, the complete feature list for each layer was not presented in the paper. The above staged and layered approaches used classifiers of the same type or of different types for detecting network intrusions. The approaches handled the attacks separately to minimize the attack categories from affecting each other in classification or detection tasks. Since every layer in Layered framework is independent, feature sets for all the four layers are not disjoint. The final goal is to improve both the attack detection accuracy and the efficiency of the system. Hence, by integrating the CRFs and the Layered Approach can build efficient and accurate single system.

V. EXPERIMENTAL METHODOLOGY

The Data Set

The data set used for the entire course of research is the DARPA KDD99 benchmark data set [4], also known as "DARPA Intrusion Detection Evaluation data set" that not only includes a large quantity of network traffic but also collects a wide variety of attacks. They setup an environment to collect TCP/IP dump from a host located on a simulated military network. Each TCP/IP connection is described by 41 discrete and continuous features and labeled as either the normal or as an attack. Attacks fall into following four main classes:

A. Denial of service (DOS)

In this type of attack an attacker makes some computing or memory resources too busy or too full to handle legitimate requests or denies legitimate users access to a machine. Examples are Apache2, Back, Land, Smurf, Teardrop.

B. Remote to user (R2L)

In this type of attack an attacker who does not have an account on a remote machine sends packets to that machine over a network and exploits some vulnerability to gain local access as a user of that machine. Examples are Dictionary, Ftp_write, Guest, Imap, Named.

C. User to root (U2R)

In this type of attacks an attacker starts out with access to a normal user account on the system and is able to exploit system vulnerabilities to gain root access to the system. Examples are Eject, Loadmo dule, Ps, Xterm, Perl.

D. Probing

In this type of attacks an attacker scans a network of computers to gather information or find known vulnerabilities. Examples are Ipsweep, Mscan, Satan, Nmap.

VI. CONCLUSION

Thus we conclude that there are various approaches and techniques to implement an intrusion detection system based on its type and mode of deployment. Each of the approaches to implement an intrusion detection system has its own advantages and disadvantages. This is apparent from the discussion of comparison among the various methods. Thus it is difficult to choose a particular method to implement an intrusion detection system over the other. This paper has drawn the conclusions on the basis of implementations performed using various techniques. New techniques keep emerging which will remove the drawbacks of the previous methods of implementation. In this paper, a new efficient and robust hybrid intrusion detection systems using conditional random field was discussed. The CRFs are very effective in improving the attack detection rate and decreasing the FAR. Feature selection and implementing the Layered framework significantly reduce the time required to train and test the model. The sequence labeling methods such as the CRFs can be very effective in detecting attacks and decreasing the false alarm rate. Compared approach with some well-known methods and found that most of the present methods for intrusion detection fail to reliably detect R2L and U2R attacks, while integrated system can effectively and efficiently detect such attacks Finally, system has the advantage that the number of layers can be increased or decreased depending upon the environment in which the system is deployed, giving flexibility to the network administrators. The areas for future research include the use of Layered CRF method for extracting features that can aid in the development of signatures for signature-based systems. This can further be extended to implement pipelining of layers in multicore processors, which is likely to result in very high performance.

Techniques	*Method*	*Parameters*	*Advantages*	*Disadvantages*
Support Vector Machine	A support vector machine constructs a hyper plane or set of hyper planes in a high or infinite dimensional space, which can be used for classification, regression or other tasks.	The effectiveness of SVM lies in the selection of kernel and soft margin parameters. For kernels, different pairs of (C, γ) values are tried and the one with the best cross-validation accuracy is picked. Trying exponentially growing sequences of C is a practical method to identify good parameters.	1. Able to model complex nonlinear decision boundaries. 2. Highly accurate. 3. Provide real-time detection capability 4. Deal with large dimensionality of data. 5. Can be used for binary-class as well as multiclass classification.	1. High algorithmic complexity and extensive memory requirements of the required quadratic programming in large-scale tasks. 2. The choice of the kernel is difficult 3. The speed is slow both in training and testing.
Clustering	The cluster with the shortest distance is selected, and if that distance is less than some constant W (cluster width) then the instance is assigned to that cluster.	Convert d based on the statistical information of the training set from which the clusters were created. 1. Let d1 be the instance after conversion. 2. Find a cluster which is closest to d1 under the metric M (i.e. a cluster in the cluster set, such that for all C1 in S, dist (C, d1) <= dist (C1, d1). 3. Classify d1 according to the label of C(either Normal or anomalous.	Can work in near linear time.	1. Observation must be numeric. 2. Consider the features independently and are unable to capture the relationship between different features of a single record which results in lower accuracy.

Artificial Neural Network	An ANN is an adaptive system that changes its structure based on external or internal information that flows through the network during the learning phase.	ANN uses the cost function *C* is an important concept in learning, as it is a measure of how far away a particular solution is from an optimal solution to the problem to be solved.	1. Able to implicitly detect complex nonlinear relationships between dependent and independent variables. 2. High tolerance to noisy data. 3. Availability of multiple training algorithms.	1. Greater computational burden. 2. Requires long training time. 3. Hard to select the best possible architecture for a neural network. 4. Require large amount of data for training.
Bayesian Method	Based on the rule, using the joint probabilities of sample observations and classes, the algorithm attempts to estimate the conditional probabilities of classes given an observation.	In Bayes, all model parameters (*i.e.*, class priors and feature probability distributions) can be approximated with relative frequencies from the training set.	1. Exhibit high accuracy and speed when applied to large databases. 2. Capability of encoding interdependences between variable and of predicating events. 3. Abiltity to incorporate both prior knowledge and data.	1. Make strict independence between the features in observations results lower attack detection accuracy. 2. Lack of available probability data. 3. A fully connected Bayesian network is complex and difficult to train. 4. 4. Higher computational effort is required.
Decision Tree	Decision tree builds a binary classification tree. Each node corresponds to a binary predicate on one attribute; one branch corresponds to the positive instances of the predicate and the other to the negative instances.	Decision Tree Induction uses parameters like a set of candidate attributes and an attribute selection method.	1. Construction does not require any domain knowledge. 2. Can handle high dimensional data. 3. Representation is easy to understand. 4. Able to process both numerical and categorical data. 5. High speed of operation and high attack detection accuracy.	1. Output attribute must be categorical. 2. Limited to one output attribute. 3. Decision tree algorithms are unstable. 4. Trees created from numeric datasets can be complex.
Hidden Markov Models	Markov chains and hidden Markov models can be used when dealing with sequential representation of audit patterns	During a first training phase, the probabilities associated to the transitions are estimated from the normal behavior of the target system. The detection of anomalies is then carried out by comparing the anomaly score (associated probability) obtained for the observed sequences with a fixed threshold	1. Modeling the ordering property of events results higher detection accuracy. 2. Effective in modeling sequences of system calls of a privileged process	1. May not always provide accurate classification as various connection level features are ignored 2. HMMs become very complex for long range dependencies in observations. 3. Results inaccuracy as the correlation among features is lost.
Layered Conditional Random Fields	Conditional Random Fields are discriminative and undirected graphical models which are used for sequence tagging. They do not make any unwarranted assumptions about the data.	• For training: – Forward Backward algorithm is used which has a complexity of O(K2T), where K is the number of states and T is the length of the sequence	1. CRF do not assume observation features to be independent 2. Not prohibitively expensive in testing. 3. CRF training is feasible for many real-world. 4. Integrated system (CRF & Layered) achieves	1. Computational expense of training. 2. Complete list of features for each level is not available.

		• For testing: – Viterbi algorithm is used which also has the same complexity	significant improvement, both, in the time required to train and test the system and also in the attack detection accuracy (F-value). 5. CRFs are robust to noise in training data. 6. CRFs avoid the label bias problem. 7. CRFs avoid a fundamental limitation of maximum entropy Markov models (MEMMs).	

VII. REFERENCES

[1] SANS Institute—Intrusion Detection FAQ, http://www.sans.org/ resources/idfaq/, 2010.

[2] Autonomous Agents for Intrusion Detection, http://www.cerias.purdue.edu/research/aafid/, 2010.

[3] CRF++: Yet Another CRF Toolkit, http://crfpp.sourceforge.net/,2010.

[4] KDD Cup 1999 Intrusion Detection Data, http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html, 2010.

[5] Overview of Attack Trends, http://www.cert.org/archive/pdf/ attack_trends.pdf, 2002.

[6] Kapil Kumar Gupta, Baikunth Nath, Ramamohanarao Kotagiri, "Layered Approach Using Conditional Random Fields for Intrusion Detection," IEEE Transactions on Dependable and Secure Computing (vol. 7 no. 1), pp. 3 5-49, 2010.

[7] J.P. Anderson, Computer Security Threat Monitoring and Surveillance, http://csrc.nist.gov/publications/history/ande80.pdf, 2010.

[8] W. Lee and S. Stolfo, "Data Mining Approaches for Intrusion Detection," Proc. Seventh USENIX Security Symp. (Security '98), pp. 79-94, 1998.

[9] H. Shah, J. Undercoffer, and A. Joshi, "Fuzzy Clustering for Intrusion Detection," Proc. 12th IEEE Int'l Conf. Fuzzy Systems (FUZZ-IEEE '03), vol. 2, pp. 1274-1278, 2003.

[10] C. Kruegel, D. Mutz, W. Robertson, and F. Valeur, "Bayesian Event Classification for Intrusion Detection," Proc. 19th Ann. Computer Security Applications Conf. (ACSAC '03), pp. 14-23, 2003.

[11] N.B. Amor, S. Benferhat, and Z. Elouedi, "Naive Bayes vs. Decision Trees in Intrusion Detection Systems," Proc. ACM Symp. Applied Computing (SAC '04), pp. 420-424, 2004.[2] W. Lee and S. Stolfo, "Data Mining Approaches for Intrusion Detection," Proc. Seventh USENIX Security Symp. (Security '98), pp. 79-94, 1998.

[12] H. Debar, M. Becke, and D. Siboni, "A Neural Network Component for an Intrusion Detection System," Proc. IEEE Symp. Research in Security and Privacy (RSP '92), pp. 240- 250, 1992.

[13] Y. Du, H. Wang, and Y. Pang, "A Hidden Markov Models-Based Anomaly Intrusion Detection Method," Proc. Fifth World Congress on Intelligent Control and Automation (WCICA '04), vol. 5, pp. 4348-4351, 2004.

[14] A. McCallum, "Efficiently Inducing Features of Conditional Random Fields," Proc. 19th Ann. Conf. Uncertainty in Artificial Intelligence (UAI '03), pp. 403-410, 2003.

[15] J. Lafferty, A. McCallum, and F. Pereira, "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data," Proc. 18th Int'l Conf. Machine Learning (ICML '01), pp. 282-289, 2001.

[16] A. McCallum, D. Freitag, and F. Pereira, "Maximum Entropy Markov Models for Information Extraction and Segmentation," Proc. 17th Int'l Conf. Machine Learning (ICML '00), pp. 591-598,2000.

[17] D.S. Kim and J.S. Park, "Network-Based Intrusion Detection with Support Vector Machines," Proc. Information Networking, networking Technologies for Enhanced Internet Services Int'l Conf. (ICOIN '03),pp. 747-756, 2003.

[18] C. Sutton and A. McCallum, "An Introduction to Conditional Random Fields for Relational Learning," Introduction to Statistical Relational Learning, 2006.